PRINCETON UNIVERSITY COS 522: COMPUTATIONAL COMPLEXITY

Homework 4

Out: *April 25* Due: *May 2*

You can collaborate with your classmates, but be sure to list your collaborators with your answer. Also, limit your answers to one handwritten page, preferably less —you just need to give enough detail to convince me. If some question doesn't seem correct or solvable, indicate why.

§1 If $s(n)q(n) \leq n$ and $r(n) \geq \log n$ then show that $\text{PCP}(r(n), q(n)) \subseteq \text{sPCP}(r(n), 1, \text{poly}(q(n)))$. (Hint: Change the PCP verifier so that it expects the proof string to be a polynomial extension of the original proof.)

§2 This question covers *list-decodability* of codes. If Σ is an alphabet, then the *distance* between two words $x, y \in \Sigma^m$, denoted $\delta(x, y)$ is the fraction of indices they differ on. (The *agreement* is $1 - \delta(x, y)$.) A *code* over Σ^m is some $\mathcal{C} \subseteq \Sigma^m$; its elements are called codewords. The minimum distance $\delta_{min}(\mathcal{C})$ is the smallest distance between two distinct codewords.

(a) Let m be a power of 2. Show that if $\mathcal{C}$ is the linear function code over $GF(2)^m$ then for any $x \in GF(2)^m$, the number of codewords that have distance at most $1/2 - \epsilon$ to x is at most $1/4\epsilon^2$. (Hint: Use the fourier transform.)

(b) Let q be a prime and $d < m < q$. Let $a_1, a_2, \ldots, a_m \in GF(q)$. *Reed Solomon* code over $GF(q)^m$ contains, for every polynomial p of degree at most d, the codeword $(p(a_1), p(a_2), \ldots, p(a_m))$. Note that $\delta_{min} = 1 - d/m$.

Let $\epsilon > 2\sqrt{d/m}$. Show that for any word x the number of codewords that have agreement at least ϵ with x is at most $2/\epsilon$.

(c) The code in (a) is inefficient since it encodes $\log m$ bits with m bits. The code in (b) is efficient (say if $m = \text{poly}(d)$) but uses a nonbinary alphabet. Consider the binary code obtained by encoding each letter of the codewords in (b) using the code in (a). (This is called *code concatenation*.) Show that this gives an efficient binary code with a list-decoding property.

§3 For this question, assume Sudan's result, which gives a randomized polynomial-time algorithm for list-decoding for the Reed-Solomon code.

Show that if some $L \in$ EXPTIME does not have circuits of size $S(n)$ then there is some $c > 0$ and a language $L' \in$ EXPTIME that is $(S(n)^c, 1/n)$-hard. (Hint: Use the multivariate polynomial extension of the characteristic function of L, and use the polynomial reconstruction procedure used in the PCP Theorem.)

Note that this result (together with Yao's XOR lemma) allows us to prove derandomization results using assumptions about worst-case complexity.

§4 A family H of hash functions $h : [N] \to [L]$ has *collision error* δ if for any $x_1 \neq x_2 \in [N]$, $\Pr_{h \in H}[h(x_1) = h(x_2)] \leq (1 + \delta)/L$.

Define $Ext : [N] \times H \to [L] \times H$ by $Ext(x, h) = h(x) \circ h$ where $\circ$ denotes concatenation. Show that if x is drawn from a distribution with min-entropy at least $\log L/\delta$ then the output is $O(\sqrt{\delta})$-pseudorandom.

(Hint: Let the collision probability of a distribution $\{p_x\}$ be $\sum_x p_x^2$. Show that (a) the collision probability is at most (min-entropy)$^{-1}$, and (b) if the collision probability of is close to that of the uniform distribution, then the distribution is close to random.)

§5 Compute the Fourier transform of the function $f : \{-1,1\}^{2k+1} \to \{-1,1\}$ defined as $f(x) =$ majority bit in x.